Measuring Software Developers' Perceived Difficulty With Biometric Sensors

Müller, Sebastian C

DOI: https://doi.org/10.1109/ICSE.2015.284

Measuring Software Developers' Perceived Difficulty With Biometric Sensors

Sebastian C. Müller
Department of Informatics, University of Zurich, Switzerland
Email: smueller@ifi.uzh.ch; *Web:* http://seal.ifi.uzh.ch/mueller; *Advisor:* Prof. Thomas Fritz (fritz@ifi.uzh.ch)

***Abstract*—As a developer works on a change task, he or she might perceive some parts of the task as easy and other parts as being very difficult. Currently, little is known about when a developer experiences different difficulty levels, although being able to assess these difficulty levels would be helpful for many reasons. For instance, a developer's perceived difficulty might be used to determine the likelihood of a bug being introduced into the code or the quality of the code a developer is working with. In psychology, biometric measurements, such as electro-dermal activity or heart rate, have already been extensively used to assess a person's mental state and emotions, but only little research has been conducted to investigate how these sensors can be used in the context of software engineering. In our research we want to take advantage of the insights gained in these psychological studies and investigate whether such biometric sensors can be used to measure developers' perceived difficulty while working on a change task and support them in their work.**

I. Problem Statement

Not all parts of a change task on which a developer works are at an equal level of difficulty. As a developer works on a change task, the developer might perceive some parts of the task as easy and other parts as having a high level of difficulty. Currently, very little is known about when a developer experiences different difficulty levels, particularly since difficulty has largely been investigated in terms of the artefacts on which a developer works. For instance, approaches have used code or process metrics to determine the complexity or defect-proneness of code [1–3]. The perceived level of difficulty might be assessed using results from psychology that indicate that biometric measurements, such as electro-dermal activity (EDA) or skin temperature can be used to measure a person's emotions and cognitive states (*eg.* [4–6]) and might thereby approximate the perceived difficulty.

Knowing what kind of emotions and difficulty levels a developer experiences while working on a change task opens up opportunities to support developers in their work. For instance, the perceived difficulty, based on biometric sensor data, might be used to determine the likelihood of a bug being introduced into the code or the quality of the code a developer is working with. As another example, the perceived emotions might be used to assess whether a developer is currently stuck on a particular difficult part of the code.

II. Related Work

Related work can broadly be categorised into three major areas: 1) research in software engineering on developers' emotions and task difficulty, 2) studies investigating the use of biometric measures in psychology, and 3) the use of biometrics in software engineering.

A. Emotions & Difficulties

Several studies observed developers or induced certain moods to investigate developers' emotions and whether they are correlated with productivity (*eg.* [7–10]). For instance, Khan *et al.* [9] induced developers' moods through videos and physical exercises to examine the influence of developers' moods on their debugging performance. These studies provide initial evidence that a correlation exists between a developer's emotions and his or her productivity.

Research also investigated ways to measure task difficulty. These approaches predominantly use various code metrics such as complexity metrics (*eg.* [1, 2]), or size metrics [11] to assess the difficulty of code comprehension tasks. Another body of research focused on empirical studies and reported on common difficulties that developers face during code comprehension tasks (*eg.* [12, 13]). None of these approaches used biometric sensors to measure task difficulty.

B. Biometrics & Psychology

In psychology, extensive research on biometrics, in particular brain-, heart-, skin-, and eye-related measurements and their relation to cognitive processes and states in humans has been conducted. For instance, brain-related measurements, such as brainwave frequency bands, are most commonly captured with an electroencephalography (EEG), and were linked to working load memory (*eg.* [14, 15]) and task engagement (*eg.* [16, 17]). Studies investigating heart-related measurements have found that the heart rate (HR) and the heart rate variability (HRV) correlate with task difficulty levels (*eg.* [18, 19]) and that the blood volume pulse (BVP) can be used to detect emotions (*eg.* [20, 21]). Skin-related measurements, such as electro-dermal activity (EDA), or the skin temperature have been linked to cognitive load and task difficulty levels (*eg.* [4, 22]) and were used to analyse emotions (*eg.* [5, 6]). Concerning eye-related measurements, research has found correlations between pupil sizes and memory load [23], cognitive workload [24], as well as task difficulty [23]. In our research, we build upon these findings and investigate how biometric sensors can be used to measure developers' emotions and difficulty levels while working on a change task.

C. Biometrics in Software Engineering

Most of the studies in software engineering using biometrics have focused on eye-tracking technology and have investigated how developers comprehend code (*eg.* [25–27]). Only very few studies investigated the use of other biometric sensors. For instance, Parnin *et al.* [28] relied on electromyography to measure developers' sub-vocal utterances and found that these utterances might be used to measure programming task difficulty. Sigmund *et al.* [29] examined the active brain regions during small code comprehension tasks using fMRI technology. In contrast to these studies, we investigate the use of a combination of biometric sensors to measure developers' emotions and difficulty while working on a change task.

III. Proposed Research and Evaluation

In our work, we take advantage of the advances and insights gained in various psychological studies and explore the potential of applying biometric sensors to the software engineering domain. In particular, we conduct exploratory studies to investigate how biometric sensors can be used to measure a developer's perceived difficulty while working on a change task. The hypothesis for our research is:

Hypothesis. *Biometric sensors can be used to measure the difficulty a developer experiences while working on a change task which in turn can be used to support developers in their work.*

To evaluate our hypothesis, we focus on the following three research questions:

Research Question 1. *Can we use biometric measurements to accurately predict whether small code comprehension tasks are difficult or easy for an individual developer?*

Research Question 2. *Can biometric sensors be used to assess the change in the emotions and progress that a developer experiences while working on a change task?*

Research Question 3. *Can we use biometric sensors to model a developer's perceived difficulty while working on code elements?*

To answer these research questions, we either already conducted or will conduct studies in which we collect biometric data from software developers performing various programming tasks. Figure 1 provides an overview of the data collection and analysis process we use in each of these studies. While the developers are working on their tasks, we record their biometric data with three different off-the-shelf sensors: an eye tracker called Eyetribe, a Neurosky Mindband EEG sensor, and an Empatica E3 wrist band. We have chosen these particular sensors for various reasons: 1) existing literature and research has linked the measurements recorded with these sensors to cognitive states and process, as well as emotions, 2) these sensors are less invasive than other similar devices, and 3) the sensors are affordable for an individual developer. Based on existing research and the results of our own studies, we clean the recorded biometric data and apply normalisation techniques to it. Afterwards, relevant features are extracted from the captured data. These features are fed into a machine learning classifier to infer measures for a developer's perceived difficulty.

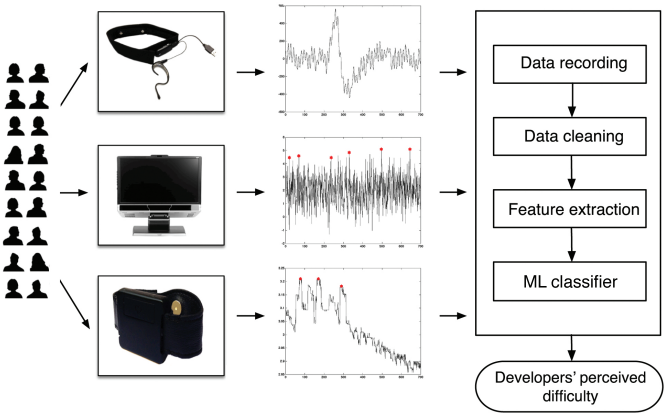

Fig. 1. Overview about the approach we follow to record biometric data and use machine learning to find measures for a developer's perceived difficultly.

This approach is used in all the studies we conducted or will conduct to answer our research questions. Each study is described in more detail in the following chapters.

A. RQ1: Biometrics & Developers' Difficulty Levels

Goal. In a first step, we investigated which biometric measurements can be used to predict whether short code comprehension tasks are perceived as easy or difficult by an individual software developer. More details on this study can be found in [30].

Study Setup. In a user study, we had 15 professional software developers work on 8 different short code comprehension tasks that were constructed to have a varying level of difficulty. While the developers where working on these tasks, we captured their biometric signals with three different sensors. After completing all the 8 tasks, the developers were asked to rank the tasks based on their difficulty.

Evaluation & Results. We examined whether we can use the captured biometric data to build a machine learning classifier that is able to predict whether a given code comprehension task was perceived as easy or difficult by an individual developer. Using a leave-one-out cross validation approach, we found that we can predict task difficulty for a developer that was not in the training set with 64.99% precision and for a task not yet trained on with 84.38% precision. This suggests that it might be possible to use biometric sensors to predict a developer's perceived difficulty while working on a small task.

B. RQ 2: Biometrics & Developers' Emotions

Goal. In a second study, we focused on longer and more realistic tasks to investigate whether biometric sensors can be used to assess the changes in emotions and progress that developers experience while working on a change task. Details on this study can be found in [31].

Study Setup. We conducted a lab study with 17 participants who worked on two different change tasks each. Again, while

the developers were working on these tasks, we collected biometric measurements with three different sensors and periodically asked the participants to assess their perceived emotions and progress on the task using a small survey.

Evaluation & Results. The results show that developers generally experience a wide range of emotions and that these emotions are in general positively correlated with the perceived progress on the change task. The analysis has also shown that it is possible to build a machine learning classifier that is able to distinguish between positive and negative emotions in 71.36% and between low and high progress in 67.70% of all cases. This provides initial evidence that biometric sensors might be used to assess a developer's perceived emotions and progress.

C. RQ 3: Modeling Difficulty to Support Developers

Goal. For the third research question, we aim to use the insights gained in the studies for the first and second research question to investigate whether we can develop a model that captures the perceived difficulty of a developer working on a change task. In a second step, we aim to examine whether this model can be used to support developers in their work, by automatically identifying code elements that have a low quality and would therefore benefit from a refactoring or a thorough code review.

Study Setup. To find answers to this research question, we plan to conduct a field study with professional software developers. While the developers are working on their usual change tasks in their normal work environment, we plan to collect two different kind of data: 1) biometric data collected with various sensors, and 2) the code elements a developer works with, captured with an eye tracker. Based on accurate time stamps that are captured with each data point, we will associated the experienced difficulty and emotions with the code elements a developer worked with.

Evaluation & Results. For the evaluation, we will investigate whether our model of perceived difficulty and emotions could be used as an indicator for code with low quality. The basic assumption is that code elements that were frequently associated with negative emotions and high perceived difficulty might lack quality compared to other code elements. To verify this, we will compare the classification of our model with the actual quality issues that were found during a code review of the change tasks the developers were working on. In a second evaluation step, we will compare our model to established quality metrics, such as complexity metrics [32].

IV. Threats & Limitations

There are several threats and limitations to our user studies. A major threat is that biometric signals can be influenced by many other factors than the difficulty or emotions a developer experiences, for instance the time of day or personality traits. To mitigate this risk, we carefully design our user studies, for instance by conducting them in a quiet environment or by investigating the personality traits of the study participants in advance. Biometric sensors are also prone to noise and the data captured with these sensors might have a high variability between individuals [33]. This requires baseline and normalisation techniques to be applied before the data can be analysed. To capture biometric data, we have to get developers to wear several sensor devices which might be considered invasive by some of them. To mitigate this risk, we have carefully chosen the sensor devices to be as non-invasive as possible. Finally, due to the small sampling size, the results found in our research might not be generalisable to other populations than the study participants or to other tasks than the ones used in the studies. We mitigate this risk by choosing study participants with various backgrounds and by selecting study tasks representative of actual change tasks. Further studies need to be conducted to investigate the generalizability of our initial results.

V. Expected Contributions

The goal of our research is to investigate, in exploratory studies, the use of biometric measures to assess a developer's perceived difficulty working on a change task. Furthermore, we examine whether these measures can be used to support developers in their work, by automatically identifying code with low quality. The expected contributions of this research proposal are threefold:

i) Initial evidence from several studies with professional software developers on the use of biometric sensors for assessing the difficulty and the emotions that a software developer perceives while working on a change task,
ii) a reusable framework for recording, cleaning, and analysing biometric measurements, and
iii) a model to capture and identify the difficulty and emotions a developer experiences and that can be used to support a developer while working on a change task.

VI. Progress & Outlook

Figure 2 provides a chronological overview about the three research questions we aim to investigate and the publication goal for each of these research questions.

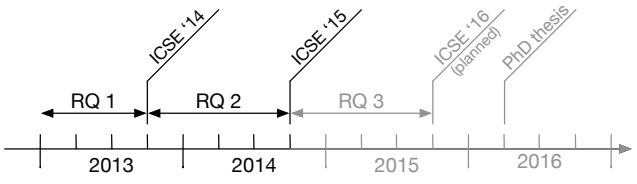

Fig. 2. Chronological overview about the research we did, the research questions we are going to answer and the publication target for each research question.

The studies for the first and second research question have already been conducted and our results and contributions have been published and presented at ICSE'14 [30], respectively ICSE'15 [31]. As a next step, we plan to conduct a further user study in the beginning of 2015 to find answers to the third research question. We will analyse the collected data during

the second quarter of 2015 and submit our contributions and results to the International Conference on Software Engineering (ICSE) 2016.

References

[1] N. Kasto and J. Whalley, "Measuring the difficulty of code comprehension tasks using software metrics," in *Australasian Computing Education Conference*, 2013.

[2] B. Katzmarski and R. Koschke, "Program complexity metrics and programmer opinions," in *ICPC*, 2012.

[3] N. Nagappan and T. Ball, "Use of relative code churn measures to predict system defect density," in *ICSE*, 2005.

[4] Y. Shi, N. Ruiz, R. Taib, E. Choi, and F. Chen, "Galvanic skin response (GSR) as an index of cognitive load," in *Ext. Abstracts on Human Fact. in Comp. Systems*, 2007.

[5] C. Collet, E. Vernet-Maury, G. Delhomme, and A. Dittmar, "Autonomic nervous system response patterns specificity to basic emotions," *Journal of the Autonomic Nervous System*, 1997.

[6] P. Ekman, R. Levenson, and W. Friesen, "Autonomic nervous system activity distinguishes among emotions," *Science*, 1983.

[7] T. Shaw, "The emotions of systems developers: An empirical study of affective events theory," in *Conference on Computer Personnel Research: Careers, Culture, and Ethics in a Networked Environment*, 2004.

[8] X. W. Daniel Graziotin and P. Abrahamsson, "Are happy developers more productive? the correlation of affective states of software developers ane their-self-assessed productivity," in *Intern. Conf. on Product-Focused Software Process Improvement*, 2013.

[9] I. A. Khan, W.-P. Brinkman, and R. M. Hierons, "Do moods affect programmers' debug performance?" *Cognition, Technology & Work*, 2011.

[10] R. H. Iftikhar Ahmed Khan, Willem-Paul Brinkman, "Towards estimating computer users' mood from interaction behaviour with keyboard and mouse," *Frontiers of Computer Science*, 2013.

[11] J. Feigenspan, S. Apel, J. Liebig, and C. Kastner, "Exploring software measures to assess program comprehension," in *ESEM*, 2011.

[12] A. Karahasanović, A. K. Levine, and R. Thomas, "Comprehension strategies and difficulties in maintaining object-oriented systems: An explorative study," *Journal of Systems and Software*, 2007.

[13] R. Tiarks and T. Roehm, "Challenges in program comprehension," *Softwaretechnik-Trends*, 2012.

[14] M. E. Smith and A. Gevins, "Neurophysiologic monitoring of mental workload and fatigue during operation of a flight simulator," in *Defense and Security*, 2005.

[15] M. Sterman, C. Mann, and D. Kaiser, "Quantitative EEG patterns of differential in-flight workload," in *Workshop on Space Operations Applications and Research*, 1993.

[16] A. E. Kramer, "Physiological metrics of mental workload: A review of recent progress," 1990.

[17] C. Berka, D. Levendowski, M. Lumicao, A. Yau, G. Davis, V. Zivkovic, R. Olmstead, P. Tremoulet, and P. Craven, "EEG correlates of task engagement and mental workload in vigilance, learning, and memory tasks," *Aviation, space, and environmental medicine*, 2007.

[18] G. F. Wilson, "Applied use of cardiac and respiration measures: Practical considerations and precautions," *Biological Psychology*, 1992.

[19] A. Rieger, R. Stoll, S. Kreuzfeld, K. Behrens, and M. Weippert, "Heart rate and heart rate variability as indirect markers of surgeons' intraoperative stress," *Intern. Archives of Occupational and Environm. Health*, 2014.

[20] R. W. Picard, E. Vyzas, and J. Healey, "Toward machine emotional intelligence: Analysis of affective physiological state," *Trans. on Pattern Analysis and Machine Intelligence*, 2001.

[21] S. D. Kreibig, F. H. Wilhelm, W. T. Roth, and J. J. Gross, "Cardiovascular, electrodermal, and respiratory response patterns to fear- and sadness-inducing films," *Psychophysiology*, 2007.

[22] N. Nourbakhsh, Y. Wang, F. Chen, and R. A. Calvo, "Using galvanic skin response for cognitive load measurement in arithmetic and reading tasks," in *Australian Computer-Human Interaction Conference*, 2012.

[23] J. Beatty, "Task-evoked pupillary responses, processing load, and the structure of processing resources," *Psychological Bulletin*, 1982.

[24] J. Klingner, "Fixation-aligned pupillary response averaging," in *Symp. on Eye-Tracking Research & Appl.*, 2010.

[25] R. Bednarik and M. Tukiainen, "An eye-tracking methodology for characterizing program comprehension processes," in *Symp. on Eye-Tracking Research & Appl.*, 2006.

[26] M. Crosby and J. Stelovsky, "How do we read algorithms? A case study," *Computer*, 1990.

[27] B. Sharif and J. Maletic, "An eye tracking study on camelcase and under_score identifier styles," in *ICPC*, 2010.

[28] C. Parnin, "Subvocalization - toward hearing the inner thoughts of developers," in *ICPC*, 2011.

[29] J. Siegmund, C. Kästner, S. Apel, C. Parnin, A. Bethmann, T. Leich, G. Saake, and A. Brechmann, "Understanding understanding source code with functional magnetic resonance imaging," in *ICSE*, 2014.

[30] T. Fritz, A. Begel, S. Müller, S. Yigit-Elliot, and M. Züger, "Using psycho-physiological measures to assess task difficulty in software development," in *ICSE*, 2014.

[31] S. C. Müller and T. Fritz, "Stuck and frustrated or in flow and happy: Sensing developers' emotions and progress," in *ICSE*, 2015.

[32] T. Menzies, J. Greenwald, and A. Frank, "Data mining static code attributes to learn defect predictors," *Trans. on Software Engineering*, 2007.

[33] R. Mandryk, *Game Usability*. CRC Press, 2008, ch. Physiological Measures for Game Evaluation.